**Amendments to the Claims:**

Please amend Claims 22 and 28 and add new Claims 29-35 as follows:

1. (previously presented) A method of processing spread spectrum signals comprising:

receiving a continuous signal ($S_{IF}$) of a comparatively high frequency;

sampling the continuous signal ($S_{IF}$) at a basic sampling rate ($r_s$), whereby a resulting sequence of time discrete signal samples ($S[s_i]$) is produced;

quantizing each signal sample into a corresponding level-discrete sample value;

forming of a plurality of data words (d(l), ..., d(N)) which each includes one or more consecutive sample values ($s_1$, ..., $S_n$); and

correlating between information in the data words (d(k)) and at least one representation (CS(i)) of a signal source specific code sequence (CS), the method comprising preparing for the correlation step, wherein, before receiving the continuous signal ($S_{IF}$), a multitude of code vectors (CV; $CV_m$) are pre-generated, each code vector representing a particular code sequence (CS(i)) of the at least one signal source specific code sequence (CS), and

the correlating step involving multiplying at least each vector in a sub-group ($CV_{m-E}$, $CV_{m-P}$; $CV_{m-L}$) of the code vectors (CV; $CV_m$) with at least one vector ($S_{IF-I}(k)$; $S_{IF-Q}(k)$) derived from the data word (d(k)).

2. (previously presented) A method according to claim 1, wherein each code vector (CV; $CV_m$) represents a particular signal source specific code sequence (CS(i)) being sampled at the basic sampling rate ($r_s$) and quantized with the quantizing process being used to produce the level-discrete sample values ($S_1$, ..., $S_n$).

3. (previously presented) A method according to claim 1, wherein the at least one signal source specific code sequence (CS) is pseudo random noise.

4. (previously presented) A method according to claim 1, wherein the receiving step involving down conversion of an incoming high-frequency signal ($S_{HF}$) to an intermediate frequency signal ($S_{IF}$), the high-frequency signal ($S_{HF}$) having a spectrum which is symmetric around a first frequency ($f_{HF}$) and the intermediate frequency signal ($S_{IF}$) having a spectrum which is symmetric around a second frequency ($f_{IF}$) being considerably lower than the first frequency ($f_{HF}$).

5. (previously presented) A method according to claim 4, further comprising the steps of:

determining a maximum frequency variation ($f_D$) of the second frequency ($f_{IF}$) due to Doppler-effects;

defining a Doppler frequency interval ($f_{IF-min}$ - $f_{IF-max}$) around the second frequency ($f_{IF}$), the Doppler frequency interval ($f_{IF-min}$, $f_{IF-max}$) having a lowest frequency limit ($f_{IF-min}$) equal to the difference between the second frequency ($f_{IF}$) and the maximum frequency variation ($f_D$), and a highest frequency limit ($f_{IF-max}$) equal to the sum of the second frequency ($f_{IF}$) and the maximum frequency variation ($f_D$);

dividing the Doppler frequency interval ($f_{IF-min}$ - $f_{IF-max}$) into an integer number of equidistant ($\Delta f$) frequency steps ($f_{IF-min}$, $f_{IF-min}+\Delta f$, ..., $f_{IF-max}$) ; and

defining a frequency candidate vector ($f_{IF-C}$) for each frequency step ($f_{IF-min}$, $f_{IF-min}+\Delta f$, ..., $f_{IF-max}$).

6. (previously presented) A method according to claim 5 further comprising the steps of:

determining an integer number of initial phase positions ($\varphi_0$, ..., $\varphi_7$) for the frequency candidate vector ($f_{IF-C}$), and

defining a carrier frequency-phase candidate vector ($V_{f\varphi}(f_{IF-C}, \varphi_C)$) for each combination of carrier frequency candidate vector ($f_{IF-C}$) and initial phase position ($\varphi_0$, ..., $\varphi_7$).

7. (previously presented) A method according to claim 6 further comprising the steps of:

determining the number of elements ($s_n$) in each carrier frequency-phase candidate vector ($V_{f\varphi}(f_{IF\text{-}C}, \varphi_C)$); and

storing the carrier frequency-phase candidate vector ($V_{f\varphi}(f_{IF\text{-}C}, \varphi_C)$) according to a data format being adapted to performing multiplication operations between the data word ($d(k)$) and a segment of a carrier frequency-phase candidate vector ($V_{f\varphi}(f_{IF\text{-}C}, \varphi_C)$).

8. (previously presented) A method according to claim 7, wherein adapting of the data format involves adding at least one element to each segment of the carrier frequency-phase candidate vector ($V_{f\varphi}(f_{IF\text{-}C}, \varphi_C)$) such that the segment attains a number of elements which is equal to the number of elements in the data word ($d(k)$), thus enabling a segment of the carrier frequency-phase candidate vector ($V_{f\varphi}(f_{IF\text{-}C}, \varphi_C)$) and one of the at least one vector ($S_{IF\text{-}I}(k)$; $S_{IF\text{-}Q}(k)$) to be processed jointly by at least one of a SIMD-operation (multiple-bits multiplications) and an XOR-operation (1-bit multiplications).

9. (previously presented) A method according claim 1 further comprising the steps of:

determining a maximum variation a code rate due to Doppler-effects, defining a Doppler rate interval ($CR_{C\text{-}min}$ - $CR_{c\text{-}max}$) around a center code rate ($CR_c$), the Doppler frequency interval having a lowest. code rate limit ($CR_{c\text{-}min}$) equal to the difference between the center code rate ($CR_c$) and the maximum code rate variation ($CR_D$), and a highest frequency limit ($CR_{c\text{-}max}$) equal to the sum of the center code rate ($CR_c$) and the maximum code rate variation ($CR_D$);,

dividing the Doppler rate interval ($CR_{C\text{-}min}$ - $CR_{c\text{-}max}$) into an integer number of equidistant ($\Delta CR$) code rate steps ($CR_{C\text{-}min}$, $CR_{C\text{-}min}+\Delta CR$, …, $CR_{c\text{-}max}$); and

defining a code rate candidate ($CR_{c\text{-}c}$) for each code rate step ($CR_{C\text{-}min}$, $CR_{C\text{-}min}+\Delta CR$, …, $CR_{c\text{-}max}$).

10. (previously presented) A method according to claim 9 further comprising determining an integer number of possible initial code phase positions (0.0, ..., 0.9) for each code rate candidate ($CR_{c\text{-}c}$).

11. (previously presented) A method according to claim 10 further comprising defining, for each signal source specific code sequence (CS(i)), a set of combinations between code rate candidate ($CR_{c-c}$) and code phase position (0.0, ..., 0.9), each combination representing a code rate-phase candidate vector.

12. (previously presented) A method according to claim 11 further comprising generating, for each signal source specific code sequence (CS(i)), a set of code vectors (CV) by:

sampling each code rate-phase candidate vector with the basic sampling rate ($r_s$) whereby a corresponding code vector (CV) is produced.

13. (previously presented) A method according to claim 1 further comprising generating a modified code vector ($CV_m$) on basis of each code vector (CV) by:

copying a particular number of elements ($E_e$) from the end of an original code vector (CV) to the beginning of the modified code vector ($CV_m$); and

copying the particular number of elements ($E_b$) from the beginning of the original code vector (CV) to the end of the code vector ($CV_m$).

14. (previously presented) A method according to claim 13 further comprising storing, for each signal source specific code sequence (CS(i)), a set of modified code vectors ($\{CV_m(CR_{c-c}, Cph)\}$), where

each modified code vector ($CV_m$) contains a number of elements ($s_l$, ..., $s_m$) representing a sampled version of at least one full code sequence (CS),

a particular modified code vector ($CV_m$) is defined for each combination of code rate candidate ($CR_{c-c}$) and code phase position (Cph).

15. (previously presented) A method according to claim 14 further comprising adapting the data format of the modified code vectors ($CV_m$) with respect to the data format of the at least one vector ($S_{IF-I}(k)$; $S_{IF-Q}(k)$) derived from the data word (d(k)), such that a modified code vector ($CV_m$) and one of the at least one vector ($S_{IF-I}(k)$; $S_{IF-Q}(k)$) may be processed jointly by at least one of a SIMD-operation (multiple-bits multiplications) and an XOR-operation (1-bit multiplications).

16. (previously presented) A method according to claim 14 further comprising involving an initial acquisition phase and a subsequent tracking phase, wherein during the acquisition phase a set of preliminary parameters are established which are required for initiating a decoding of signals received during the tracking phase, a successful acquisition phase resulting in at least one signal source specific code sequence (CS) being identified and a transmitted signal being rendered possible to track, each of the at least one signal source specific code sequence (CS) being associated with tracking characteristics in the form of:

a modified code vector ($CV_m$),

a carrier frequency candidate vector ($f_{IF-C}$),

an initial phase position ($\varphi_c$),

a code phase position (Cph), and

a code index (Cl) denoting a starting sample value for the modified code vector ($CV_m$).

17. (previously presented) A method according to claim 16 wherein said tracking involves:

calculating, based on the tracking characteristics, a prompt pointer ($P_p$) for each modified code vector ($CV_m$), the prompt pointer ($P_p$) indicating a code sequence start position, and an initial prompt pointer ($P_p$) being equal to the code index (CI): and

assigning, around each prompt pointer ($P_p$), at least one pair of early- and late pointers ($P_E$, $P_L$; $P_{E1}$, $P_{LI}$ $P_{E2}$, $P_{L2}$), where the early pointer ($P_E$) specifies a sample value being positioned at least one element before the prompt pointer's ($P_p$) position, and the late pointer ($P_L$) specifies a sample value being positioned at least one element after the prompt pointer's ($P_p$) position.

18. (previously presented) A method according to claim 17, wherein said tracking involves:

receiving a sequence of incoming level-discrete sample values (1210),

forming data words (d(l), ..., d(N)) of the sample values (1210) such that each data word (d(k)) contains a number of elements which is equal to the number of elements ($S_n$) in each carrier frequency-phase candidate vector ($V_{f\varphi}(f_{IF-C}, \varphi_C)$); calculating a

relevant set of carrier frequency-phase candidate vectors ($V_{f\varphi}(f_{IF\text{-}C}, \varphi_C)$) for the data word ($d(k)$); and

acquiring, for each carrier frequency-phase candidate vector ($V_{f\varphi}(f_{IF\text{-}C}, \varphi_C)$) in the relevant set, a pre-generated in-phase representation ($f_{IFI}$) and a quadrature-phase representation ($f_{IFQ}$) of the vector ($V_{f\varphi}(f_{IF\text{-}C}, \varphi_C)$) respectively.

19. (previously presented) A method according to claim 18, wherein said tracking involves:

multiplying each data word ($d(k)$) with the in-phase representation ($f_{IFI}$) of the carrier frequency-phase candidate vector ($V_{f\varphi}(f_{IF\text{-}C}, \varphi_C)$), in the relevant set to produce a first intermediate-frequency-reduced information word ($S_{IF\text{-}I}(k)$), multiplying each data word ($d(k)$) with the quadrature-phase representation ($f_{IFQ}$) of the carrier frequency-phase candidate vector ($V_{f\varphi}(f_{IF\text{-}C}, \varphi_C)$) in the relevant set to produce a second intermediate-frequency-reduced information word ($S_{IF\text{-}Q}(k)$).

20. (previously presented) A method according to claim 19, wherein said tracking involves:

multiplying the first intermediate-frequency-reduced information word ($S_{IF\text{-}I}(k)$) with a modified code vector ($CV_{m\text{-}p}(k)$) starting at a position indicated by the prompt pointer ($P_p$) to produce a first prompt-despread symbol string ($\Lambda_{Ip}(k)$), multiplying the first intermediate-frequency-reduced information word ($S_{IF\text{-}I}(k)$) with a modified code vector ($CV_{m\text{-}E}(k)$) starting at a position indicated by an early pointer ($P_E$) to produce a first early-despread symbol string ($\Lambda_{IE}(k)$);

multiplying the first intermediate-frequency-reduced information word ($S_{IF\text{-}I}(k)$) with a modified code vector ($CV_{m\text{-}L}(k)$) starting at a position indicated by a late pointer ($P_L$) to produce a first late-despread symbol string ($\Lambda_{IL}(k)$) ;

multiplying the second intermediate-frequency-reduced information word ($S_{IF\text{-}Q}(k)$) with a modified code vector ($CV_{m\text{-}p}(k)$) starting at a position indicated by the prompt pointer ($P_p$) to produce a second prompt-despread symbol string ($\Lambda_{Qp}(k)$) ;

multiplying the second intermediate-frequency-reduced information word ($S_{IF\text{-}Q}(k)$) with a modified code vector ($CV_{M\text{-}E}(k)$) starting at a position indicated by the early pointer ($P_E$) to produce a second early-despread symbol string ($\Lambda_{QE}(k)$) ; and

multiplying the second intermediate-frequency-reduced information word ($S_{IF-Q}(k)$) with a modified code vector ($CV_{M-L}(k)$) starting at a position indicated by the late pointer ($P_L$) to produce a second late-despread symbol string ($\Lambda_{QL}(k)$).

21. (previously presented) A method according to claim 20, wherein said tracking involves deriving, for each despread symbol string, a resulting data word ($D_{R-Ip}(k)$, $D_{R-IE}(k)$, $D_{R-IL}(k)$, $D_{R-Qp}(k)$, $D_{R-QE}(k)$; $D_{R-QL}(k)$).

22. (currently amended) A method according to claim 21 wherein said deriving comprises deriving the resulting data words ($D_{R-Ip}(k)$, $D_{R-IE}(k)$, $D_{R-IL}(k)$, $D_{R-Qp}(k)$, $D_{R-QE}(k)$; $D_{R-QL}(k)$) by looking up a respective pre-generated value in a table ~~(1310)~~.

23. (previously presented) A method according to claim 19, wherein multiplying comprises performing the multiplication between the data word ($d(k)$) and the in-phase representation ($f_{IFI}$) of the carrier-frequency-phase candidate vector ($V_{f\varphi}(f_{IF-C}, \varphi_C)$) respective between the data work ($d(k)$) and the quadrature-phase representation ($f_{IFQ}$) of the carrier frequency-phase candidate vector ($V_{f\varphi}(f_{IF-C}, \varphi_C)$) by means of at least one of a SIMD-operation (multiple-bits multiplications) and an XOR-operation (1-bit multiplications).

24. (previously presented) A method according to claim 19, wherein multiplying comprises performing the multiplication between the intermediate-frequency-reduced information words ($S_{IF-I}(k)$; $S_{IF-Q}(k)$) and the modified code vectors ($CV_{m-p}$, $CV_{m-E}$; $CV_{m-L}$) by means of at least one of a SIMD-operation (multiple-bits multiplications) and an XOR-operation (1-bit multiplications).

25. (previously presented) A method according to claim 19 further comprising propagating, in connection with completing the processing of a current data word ($d(k)$) and initiating the processing of a subsequent data word ($d(k+l)$):

a pointer ($P_d$) indicating a first sample value of the subsequent data word ($d(k+l)$);

a group of parameters describing the relevant set of carrier frequency-phase candidate vectors ($V_{f\varphi}(f_{IF-C}, \varphi_C)$) ;

the relevant set of code vectors ($CV_m$); and

prompt-, early-, and late pointers ($P_P$, $P_E$, $P_L$).

26. (previously presented) A computer program directly loadable into the internal memory of a computer, comprising software for controlling the steps of claim 1 when said program is run on the computer.

27. (previously presented) A computer readable medium, having a program recorded thereon, where the program is to make a computer control the steps of claim 1.

28. (currently amended) A signal receiver ~~(1500)~~ for receiving navigation data signals transmitted in a navigation satellite system comprising:

a radio front end unit ~~(1510)~~ adapted to receive a continuous radio signal ($S_{HF}$) and in response thereto produce a corresponding electrical signal ($S_{IF}$) comparatively high frequency;

an interface unit ~~(1520)~~ adapted to receive the electrical signal ($S_{IF}$) and in response thereto produce a sequence of sample values being divided into data words (d(k));

a digital processor unit ~~(1530)~~ adapted to receive the data words (d(k)) and in response thereto demodulate a data signal and including a memory means; and

a computer program according to claim 16 is loaded in the memory means ~~(1535)~~.

29. (new) A software receiver comprising:

a receiver capable of receiving a radio signal;

means for digitizing the radio signal; and

a software correlator capable of mixing the digitized radio signal to form a baseband signal using bit-wise parallelism.

30. (new) The software receiver of claim 29 wherein said software correlator comprises:

means for computing correlations between the baseband signal and at least one pseudo-random number (PRN) code using the bit-wise parallelism.

31. (new) The software receiver of claim 30 wherein said software correlator further comprises:

means for computing accumulations from the correlations using the bit-wise parallelism.

32. (new) The software receiver of claim 31 further comprising:

application-specific code capable of computing navigation data using the accumulations.

33. (new) The software receiver of claim 29 wherein said means for digitizing comprises:

means for down-converting the radio signal to an intermediate frequency; and

a digitizer capable of digitizing the intermediate frequency.

34. (new) The software receiver of claim 33 wherein said digitizer produces at least one bit/sample.

35. (new) The software receiver of claim 33 wherein said digitizer is an analog to digital converter.